```
Naive Bayes Classifier : sentiment analysis on a Twitter
  dataset
# Install required packages if not already installed
 # pip intal ntk scikit-learn pandas
                                        # Make houdictions
 impost handes as he
                              y four = kno -predict (x-tot)
 insport nltk
 from skleans. model - selection import train-test - split
 6xom sklean feature extraction text inpost Count Vectorizes
 boom skleam. na ve bayes impost Multinomial NB
 from sklean metrics impost classification - report, accuracy - score
# Download stopwords if not already present men toget
ntk. download ("stop woods")
 from nltk · coxpus import stopwords
 impost string
# sample Twitter dataset (replace with your own dataset for real
  application)
  data = 3
  (tweet):
     a I love this phone!",
    "This movie is torsible.",
    "Had on awesome day today:)",
    " I hate waiting in traffic ")
     "Such a boring game",
```

" Best concert ever!",

```
" fim so sad sight now", esotron sintery of test travers the
" What a great experience [11 () regisated tood = regisated
 a worst customer service !)
"Feeling happy and blossed!" ],
                                       testob tilles
 (Sentiment):[
  (Positive), 'negative', (Positive), (negative), (negative), (Positive),
  'negative', (Positive), (negative), Positive)] }
                       + Train Naive Boyes classifier
#Load the dataset
 db = pd. Data Frame (data) (301 laimon = Jun = sailiseab - da
                     no - classifies fit (x train, y-troi
#Preferocessing function
                                   #Make predictions
del preprocess - test (text):
text = text · lower()
 text = " · join (Cchan for char in text if chan not in
  string - punduation ])
tokens = text splitu to tolor _ coto flored time
tokens = [word for word in tokens if word not in stopwords.
(coords ( english))
return ! . join (takens)
# Apply preprocessing
of [clean -tweet'] = of [tweet'] . apply (preprocess-text)
```

# convert text to feature vectors " he has a mile Vectorizer = Count Vectorizer() 1 somming took a tolk X = Vectorizer ofit \_transform (df [clean\_tweet']) y = of [ sentiment ] "Feeling hoppy and blessed I" ]. # split dateset X\_train, X\_test, y\_train, y\_test = train\_test -split (x,y, test -size = 0.2, random - state = 42) # Train Naive Bayes classifies tood the dideset nb-classifier = Multinomial NBU (stab) amos 7 and about = 10 nb - classifier. fit (x-train, y-train) #Make predictions y-pred = nb - classifies . predict (x-test) test = test · lower () # Evaluate the model print ("Accuracy:", accuracy - score (y-test, y-pred)) print (" \n classification Report:") print (classification - report (y-test, y-pred)) tokens = wood for wood in tokens if wood not in Schwas ( dailper ) sproon return ! . join (tokens)

of Colom = toot ] = of Ctoot I of the Charles was - toot

lussosoxyard higher to

indicat mathetist as hit works

8<sup>th</sup> Program Accuracy: 0.0

Classification Report:

precision recall f1-score support

negative 0.00 0.00 0.00 2.0 positive 0.00 0.00 0.00 0.00

accuracy 0.00 2.0 macro avg 0.00 0.00 0.00 2.0 weighted avg 0.00 0.00 0.00 2.0

John Mary Comment

# Visualize data before distance

Sos-scatte plot (X= (Annual Tocome (K&) ), y= Spine

( ) b = stab

Alt . Little ( Customer Distribution )

plt o shows

while I see hit I suppose of challed